

## **BACKGROUND OF THE INVENTION**

This invention relates generally to distributed computer systems that allow remote monitoring of events occurring on another computer.

The time is fast approaching where most computers are interconnected together in some type of network whether it is a local area network, wide area network or the Internet. Often a person may want to display the information, screen or data of a computer that is located in another area in real time. The other computer may be located in another room, in another building or across the country.

Applications that may be remotely monitored, but not limited to, include medical instrumentation, maintenance, technical support, and education.

It is clear from the foregoing that there is a need for a real time methodology and system to enable remote computer monitoring of display and events occurring on another computer over distributed computer networks.

## **SUMMARY OF THE INVENTION**

The present invention contemplates a system that allows a computer to be located remote from another computer that is the source of the display or events and provide real time remote monitor of that display or events over a distributed computer network.

Although the present invention is described relative to the Internet, its application is not so limited and is intended to be used on any distributed computer system in which a person interacts with another computer for the purposes of remotely monitoring events on the other computer.

In such a distributed computer network, a computer may contain a software application displaying information on a screen and a person in a remote location desires to view the same displayed information on their local computer screen in real time. This information is intended to give the remotely located person the ability to perform their job as if they were located at the source computer.

These subsystems, and other subsystems expressed or implied in this document, are accomplished in one embodiment by a system for connecting, over a distributed computer network, a remotely located computer to another computer containing source data and displaying that source data on the remote computer, the system comprising: (a) an executing software program retrieving the source display data on the server, (b) a distributed computer network for transmitting the display data, (c) an executing software program on the remote computer for receiving the transmitted display data and displaying the same data on the computer monitor.

## DRAWINGS IN BRIEF

FIGS. 1A and 1B are block diagrams of the two computer configurations used in the preferred embodiment.

FIG. 1C is a graphical representation of the preferred memory organization for the computer illustrated in FIG. 1A.

FIG. 1D graphically illustrate the connection between the two computers.

FIG. 1E is a flowchart of the preferred embodiment's remote monitoring system.

FIG. 1F is a flowchart of the preferred embodiment messaging subsystem.

## DRAWINGS IN DETAIL

FIGS. 1A and 1B are block diagrams of the two computer configurations used in the preferred embodiment.

FIG. 1A illustrates the configuration of the computer or server containing the source display and events that are going to be monitored. As shown, computer 10A is a simple layout of a Central Processing Unit (CPU) 11A which uses both non-volatile memory 12A and Random Access Memory (RAM) 13A. RAM 13A contains the currently displayed data on Monitor 16A that a operator uses to view the display data.

Communications to and from CPU 11A is via Network Interface Card 14A, which can be a modem, attached to a phone line or a card, directly connected to a network that communicates with other computers via Communications Line 15B.

Computer 10B, illustrated in FIG. 1B, shows the preferred computer configuration for the computer or client that is remotely communicating to the server 10A and displaying the source data originating from computer 10A. As shown, computer 10B is a simple layout of a Central Processing Unit (CPU) 11B which uses both non-volatile memory 12B and Random Access Memory (RAM) 13B. In the case of the remote client computer, RAM 13B contains the real time display data of RAM 13A of Computer 11A communicated via CPU 11B through NIC 14B and Communications Line 15B.

The system is able to communicate with an operator at Computer 10B using Monitor 16B that is displaying via the real time display data of RAM 13B and thus RAM 13A can be viewed on Monitor 16B.

Keyboard 17A is used to communicate operator commands to CPU 11A. In like fashion, mouse input device 18A is also used for operator input to CPU 11B. Any event change from Keyboard 17A, Mouse 18B or display data in RAM 13A causes an update to occur on the remote computer 10B and the Monitor 16B.

Optional printer 19 is used to create a hard copy of the material being displayed to the operator via Monitor 16.

The input devices for Computer 10B are the same as Computer 10A. Keyboard 17B and Mouse 18B are used to communicate operator commands to CPU 11B.

FIG. 1C is a graphical representation for the preferred memory organization for the computer illustrated in FIG. 1A.

Memory 9, located preferably in volatile memory 13A, contains the data that is displayed on Monitor 16A. It is in the form of pixel data, which contains an x and y coordinate.

Memory 10, located preferably in non-volatile memory 12A, contains data about the setup of Computer 10A and the software responsible for communicating the display data.

FIG. 1D graphically illustrates the connection of Computers 10A and 11B. A connection is initiated by Computer 11B through Network 20 to Computer 11A. Computer 11A is identified by the URL address of the computer. The operator enters in this URL address on Computer 11B and after initial handshaking authentications, the display on Monitor 16A is shown on Monitor 16B.

FIG. 1E is a flow-chart of the preferred embodiment's remote monitoring system.

After start 40A, a connection is made from the client computer to the server computer 40B over the distributed computer network using the URL associated with the server computer. The operator on the client computer enters the password 40C associated with the server computer. The server searches its memory to validate the password. If the password does not match, the client is disconnected 40F and the process is stopped 40G. If the password matches, the initialization continues.

The client is initialized 40H by the client by sending passed authentication message, followed by the version of the communications protocol used and whether the client will have exclusive access to the server or that it may be shared with other clients.

The server is initialized 40I , the width and height of the server's display memory, its pixel format and the server name are transmitted by the server.

The server then begins to communicate the display data that will be shown on the client monitor. These messages continue until the server or the client initiate a disconnect message 40K and the process is stopped 40L.

FIG. 1F illustrates the flow chart transmitted/receive event messages.